1F
2154
2700

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Patent Application

Applicant(s): Berenbaum et al.
Case: 7-2-3-3
Serial No.: 09/538,670
Filing Date: March 30, 2000
Group: 2154
Examiner: Larry D. Donaghue

I hereby certify that this paper is being deposited on this date with the U.S. Postal Service as first class mail addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450

Signature: Date: March 8, 2004

Title: Method and Apparatus for Allocating Functional Units in a Multithreaded VLIW Processor

TRANSMITTAL OF APPEAL BRIEF

Mail Stop Appeal Brief
Commissioner of Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Submitted herewith are the following documents relating to the above-identified patent application:

1. Appeal Brief (original and two copies); and
2. Copy of Notice of Appeal, filed on January 8, 2004, with copy of stamped return postcard indicating receipt of Notice by PTO on January 13, 2004.

There is an additional fee of $330 due in conjunction with this submission under 37 CFR §1.17(c). Please charge **Deposit Account No. 50-0762** the amount of $330, to cover this fee. In the event of non-payment or improper payment of a required fee, the Commissioner is authorized to charge or to credit **Deposit Account No. 50-0762** as required to correct the error. A duplicate copy of this letter and two copies of the Appeal Brief are enclosed.

Respectfully,

Date: March 8, 2004

Kevin M. Mason
Attorney for Applicant(s)
Reg. No. 36,597
Ryan, Mason & Lewis, LLP
1300 Post Road, Suite 205
Fairfield, CT 06824
(203) 255-6560

#10 RWB 3-19-04

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Patent Application

Applicant(s):	Berenbaum et al.
Case:	7-2-3-3
Serial No.:	09/538,670
Filing Date:	March 30, 2000
Group:	2154
Examiner:	Larry D. Donaghue

I hereby certify that this paper is being deposited on this date with the U.S. Postal Service as first class mail addressed to the Commissioner for Patents, P.O. 1450, Alexandria, VA 22313-1450

Signature: Jena Mannino Date: March 8, 2004

Title: Method and Apparatus for Allocating Functional Units in a Multithreaded VLIW Processor

---

APPEAL BRIEF

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

Applicants hereby appeal the final rejection dated November 14, 2003, of claims 1 through 16 of the above-identified patent application.

REAL PARTY IN INTEREST

The present application is assigned to Agere Systems Inc., as evidenced by the statement under 37 CFR 3.73 (b) submitted on March 24, 2003. The assignee, Agere Systems Inc., is the real party in interest.

RELATED APPEALS AND INTERFERENCES

A Notice of Appeal was filed on January 8, 2004 and an Appeal Brief is being submitted simultaneously, herewith, in related United States Patent Application Number 09/538,755 entitled "Method and Apparatus for Splitting Packets in a Multithreaded Very Large Instruction Word Processor," assigned to the assignee of the present invention and incorporated by reference herein.

STATUS OF CLAIMS

Claims 1 through 16 are pending in the above-identified patent application. Claims 1-16 remain rejected under 35 U.S.C. §102(b) as being anticipated by Chung et al. (United States Patent Number 5,404,469).

STATUS OF AMENDMENTS

There have been no amendments filed subsequent to the final rejection.

SUMMARY OF INVENTION

The present invention is directed to a method and apparatus for allocating functional units in a multithreaded very large instruction word (VLIW) processor. The present invention combines the techniques of conventional VLIW architectures and conventional multithreaded architectures to reduce execution time within an individual program, as well as across a workload (page 7, line 5, to page 8, line 6). The present invention utilizes a compiler to detect parallelism. The disclosed multithreaded VLIW architecture exploits program parallelism by issuing multiple instructions, in a similar manner to single threaded VLIW processors, from a single program sequencer, and also supports multiple program sequencers, as in simultaneous multithreading (page 8, line 7, to page 9, line 25).

ISSUES PRESENTED FOR REVIEW

Whether claims 1-16 are properly rejected under 35 U.S.C. §102(b) as being anticipated by Chung et al.

GROUPING OF CLAIMS

The rejected claims stand and fall together.

ARGUMENT

Independent claims 1, 5, 9, 12, 15, and 16 were rejected under 35 U.S.C. §102(b) as being anticipated by Chung et al.

The Examiner asserts that Chung teaches an allocator for selecting and forwarding the instructions to the functional units based on the priority (col. 3, line 54, to col. 4, line 63; col. 3, lines 8-29; col. 7, lines 20-40; col. 8, lines 32-55).

Applicants note that, although Chung teaches that instructions are allocated to functional units, the allocation of instructions is not done independently of the type of instruction ready for execution within each thread. Chung teaches that "the processor 100 comprises four function units FU1, FU2, FU3, FU4. Illustratively, FU1 is an arithmetic unit, FU2 is a logic unit, FU3 is a load/store unit and FU4 is a branch unit." Col. 7, lines 43-46. Thus, each functional unit is *dedicated* to executing particular types of instructions and, therefore, each functional unit can only be allocated to a thread that has an instruction ready for execution wherein the instruction type matches the capability of the functional unit. The allocation of the functional units is *dependent* on the type of instructions ready for execution within each thread. Independent claims 1, 5, 9, 12, 15, and 16 require independently allocating said functional units to any thread in said multithreaded instruction stream. This feature is supported in the specification on page 6, lines 21-23, (of the substitute specification) wherein it is disclosed that "the illustrative Multithreaded VLIW processor 600 includes nine functional units 620-1 through 620-9, which can be allocated *independently* to any thread TA-TC."

Thus, Chung et al. do not disclose or suggest independently allocating said functional units to any thread in said multithreaded instruction stream, as required by independent claims 1, 5, 9, 12, 15, and 16, as amended.

Conclusion

The rejections of the independent claims under section §102 in view of Chung et al. are therefore believed to be improper and should be withdrawn. The rejected dependent claims are believed allowable for at least the reasons identified above with respect to the independent claims.

The attention of the Examiner and the Appeal Board to this matter is appreciated.

Respectfully,

Date: March 8, 2004

Kevin M. Mason
Attorney for Applicant(s)
Reg. No. 36,597
Ryan, Mason & Lewis, LLP
1300 Post Road, Suite 205
Fairfield, CT 06824
(203) 255-6560

APPENDIX

1. A multithreaded very large instruction word processor, comprising:

a plurality of functional units for executing a plurality of instructions from an instruction stream having a plurality of threads, said threads having a priority; and

an allocator that selects instructions from said instruction stream and forwards said instructions to said plurality of functional units, said allocator selecting said instructions based on said thread priority and independently allocating said functional units to any thread in said multithreaded instruction stream.

2. The multithreaded very large instruction word processor of claim 1, wherein said thread priority allows different threads to have different priorities.

3. The multithreaded very large instruction word processor of claim 1, wherein said allocator selects and forwards said instructions for execution belonging to the thread with the highest priority.

4. The multithreaded very large instruction word processor of claim 1, wherein said allocator selects and forwards said instructions based on said thread priority and on a resource availability.

5. A multithreaded very large instruction word processor, comprising:

a plurality of functional units for executing a plurality of instructions from a multithreaded instruction stream; and

an allocator that selects instructions from said instruction stream and forwards said instructions to said plurality of functional units, said allocator selecting said instructions based on resource availability and independently allocating said functional units to any thread in said multithreaded instruction stream.

6. The multithreaded very large instruction word processor of claim 5, wherein said resource availability allows said instructions to be allocated only if the resources required by the instructions are available for the next cycle.

7. The multithreaded very large instruction word processor of claim 5, wherein said resources comprise said functional units.

8. The multithreaded very large instruction word processor of claim 5, wherein said allocator selects and forwards said instructions based on said resource availability and on a priority assigned to said threads.

9. A method of processing instructions from an instruction stream having a plurality of threads in a multithreaded very large instruction word processor, comprising the steps of:

executing said instructions using a plurality of functional units, said threads having a priority;

selecting instructions from said instruction stream based on said thread priority; and

forwarding said selected instructions to said plurality of functional units, wherein said functional units can be allocated independently to any thread in said multithreaded instruction stream.

10. The method of claim 9, wherein said thread priority allows different threads to have different priorities.

11. The method of claim 9, wherein said selection step selects said instructions for execution belonging to the thread with the highest priority.

12. A method of processing instructions from an instruction stream having a plurality of threads in a multithreaded very large instruction word processor, comprising the steps of:

executing said instructions using a plurality of functional units;

selecting instructions from said instruction stream based on resource availability; and

forwarding said selected instructions to said plurality of functional units, wherein said functional units can be allocated independently to any thread in said multithreaded instruction stream.

13. The method of claim 12, wherein said resource availability allows said instructions to be allocated only if the resources required by the instructions are available for the next cycle.

14. The method of claim 12, wherein said resources comprise said functional units.

15. An article of manufacture for processing instructions from an instruction stream having a plurality of threads in a multithreaded very large instruction word processor, comprising:

a computer readable medium having computer readable program code means embodied thereon, said computer readable program code means comprising program code means for causing a computer to:

execute said instructions using a plurality of functional units, said threads having a priority;

select instructions from said instruction stream based on said thread priority; and

forward said selected instructions to said plurality of functional units, wherein said functional units can be allocated independently to any thread in said multithreaded instruction stream.

16. An article of manufacture for processing instructions from an instruction stream having a plurality of threads in a multithreaded very large instruction word processor, comprising:

a computer readable medium having computer readable program code means embodied thereon, said computer readable program code means comprising program code means for causing a computer to:

execute said instructions using a plurality of functional units;

select instructions from said instruction stream based on resource availability; and

forward said selected instructions to said plurality of functional units, wherein said functional units can be allocated independently to any thread in said multithreaded instruction stream.